

# 움직임 정보 및 중첩 영역을 이용한 가중치 기반의 움직임 벡터 합성 기법

김현희\*, 김성민, 이승원, 정기동  
부산대학교 컴퓨터공학과  
e-mail:h2k@pusan.ac.kr

## Weight-based Motion Vector Composition using Activity Information and Overlapped Area

Hyun-Hee Kim\*, Sung-Min Kim, Seung-Won Lee, Ki-Dong Jung  
Dept of Computer Engineering, Pusan National University

### 요 약

멀티미디어 압축 및 이동 통신 기술의 발전으로 다양한 형태의 멀티미디어 서비스가 이슈화되고 있다. 비디오를 전송하기 위해서는 많은 대역폭을 필요로 하지만, 모든 네트워크가 높은 수준의 대역 및 처리 능력을 가지는 것은 아니다. 이질적인 네트워크간의 멀티미디어를 서비스하기 위해서는 네트워크 상황 또는 수신자의 처리 능력에 맞도록 재 부호화해야 하지만 그 처리비용이 높다. 트랜스코딩 기법 중에서 시간당 요구된 프레임의 개수를 조절하면 제거된 프레임의 움직임 벡터를 재 사용하여 비트율을 감소시킬 수 있다.

본 논문에서는 기존의 기법보다 향상된 움직임과 중첩 영역의 정보를 적용한 WBVC(Weight-Based Vector Composition) 기법을 제안한다. 실험을 통한 기존의 기법과의 비교 분석 결과, 비슷한 계산 복잡도에서 제안한 WBVC 기법이 높은 성능을 보였다.

### 1. 서론

최근 멀티미디어 전송에 관한 연구가 활발하게 진행되고 있다. 특히 비디오 스트리밍에 관한 응용 기술에서는 관련 기술의 발전과 더불어 다양해진 요구에 상응하는 효율적인 서비스 방법에 대한 관심이 고조되고 있다. 네트워크에는 다양한 상황이 존재하므로 각각의 유효 대역폭에 적응적이고 유연성 있는 서비스가 필요하다. 일부 실시간 응용에서는 비디오 인코더의 흐름 제어를 이용하지만, 이미 특정 비트율로 부호화된 미디어를 사용하는 응용의 경우에는 흐름 제어 방법으로 효율적인 서비스가 불가능하다.

비디오 트랜스코딩은 이미 부호화된 신호를 또 다른 신호로 변환하는 것이다[1]. 이질적인 네트워크에 차등적 혹은 적응적인 서비스를 위해서는 부호화된 비디오 스트림의 비트율 변환이 필요하다. 비디오 포맷을 변환하지 않으면서 비트율을 조절하는 기

법들 중에서, 프레임의 개수를 조절하는 기법은 트랜스코더에서 시간당 프레임의 개수를 계속적으로 줄임으로써 비트율을 감소시킨다[2].

제거된 프레임의 움직임 벡터를 사용하여 단순히 재구성하면 계산량은 줄어들지만 비디오의 화질 저하를 초래할 수 있다. 시간적 연관성을 가지는 전후 프레임들 간의 새로운 움직임 벡터는 움직임 예측 기법을 통해 정확하게 재구성될 수 있지만 높은 계산량을 요구하므로 실시간 스트리밍 서비스에 적당하지 않다. 트랜스코더에서는 프레임이 제거되더라도 해당 움직임 벡터는 재사용할 수 있으므로, 이러한 높은 계산량에 대한 해결책으로 움직임 상태 정보량 또는 중첩 영역의 크기를 기반으로 근접한 움직임 벡터를 선택하는 기법들이 연구되고 있다[3,4].

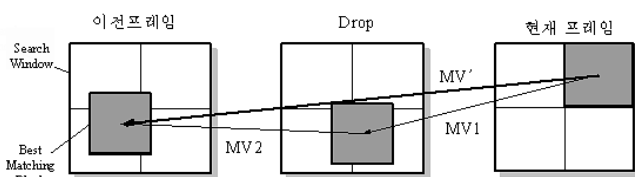
본 논문에서는 비트율을 감소하는 과정에서 제거된 프레임을 참조하고 있는 프레임의 움직임 벡터를

재구성하는 가중치 기반의 벡터 합성 기법을 제안하고, 기존의 기법들과 실험을 통해서 성능을 분석하였다. 기존의 기법들은 움직임 정보 또는 중첩 영역을 기준으로 움직임 벡터를 선택하지만, 제안하는 기법은 움직임 정보와 중첩 영역의 가중치를 기반으로 하여 제거된 프레임의 벡터를 합성하여 보다 근접한 움직임 벡터를 구성한다. 각 기법의 성능 평가의 결과는 H.263[5] QCIF(176 x 144) 비디오를 수정된 FFmpeg[6]으로 실험하여 PSNR(dB)과 계산 복잡도( $\mu s$ )로 나타냈다.

본 논문의 구성은 다음과 같다. 2장에서는 관련 연구들을 소개하고, 제안한 가중치 기반의 벡터 합성 기법과 예는 3장에서 설명한다. 4장에서는 각각의 기법들을 실험을 통하여 성능을 평가하였다. 마지막으로, 5장에서는 결론 및 향후 과제를 제시한다.

## 2. 관련 연구

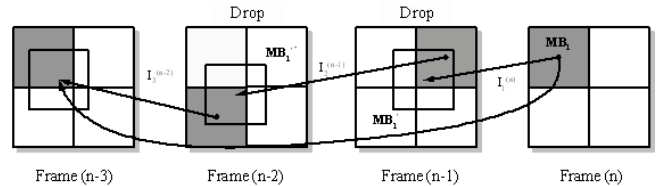
시간당 프레임의 개수를 조절하여 비트율을 감소시키는 트랜스코더에서 움직임 벡터를 새로 추출하는 것은 계산량을 높이는 주요 요인이다. [그림 1]은 하나의 프레임이 제거될 경우 각 프레임에서의 움직임 벡터들의 관계와  $MV'$ 가  $MV1$ 과  $MV2$ 의 합으로 계산된다는 것을 보여준다. 계산량은 적지만 프레임 간의 시간적 연관성이 적은 결과를 초래할 수 있다. 본 장에서는 정확한 움직임 벡터를 추출을 위해 입력된 프레임과의 중첩된 영역을 기반으로 제거된 프레임에서 움직임 벡터를 선택하는 FDVS와 활동 정보량을 기반으로 움직임 벡터를 선택하는 ADVS 기법을 소개한다.



[그림 1] 움직임 벡터의 재구성

### 2.1 FDVS(Forward Dominant Vector Selection)

제거되는 프레임에서 현재 프레임의 매크로블록이 참조하는 영역 중 가장 큰 중첩 영역을 포함하는 매크로블록이 가지는 움직임 벡터를 선택하는 기법이다[3]. 예를 들어, [그림 2]는 연속된 두 개의 프레임이 제거되는 경우의 움직임 벡터 재구성 방법을 보여주고 있다. 프레임(n-1)이 제거되었을 경우  $MB_1$ 과 중첩되는 영역이 가장 큰 오른쪽 상단의 매크로블록이 선택되었고, 제거되는 프레임이 (n-2)일 때에는 좌측 하단의 매크로블록이 선택된 것을 알 수 있다. 결국,  $MB_1$ 의 재구성된 움직임 벡터는  $MV' = I_1^{(n)} + I_2^{(n-1)} + I_3^{(n-2)}$ 와 같다. 벡터 탐색 기법으로 HAVS를 사용한다[3].



[그림 2] FDVS 기법의 움직임 벡터 재구성

### 2.2 ADVS(Activity Dominant Vector Selection)

일반적으로 활동량은 양자화를 거친 블록에 포함된 0이 아닌 DCT 계수의 개수에 비례한다. ADVS 기법은 중첩된 영역에서 매크로블록이 아닌  $8 \times 8$  블록의 0이 아닌 DCT 계수의 개수를 측정하여 그 개수가 가장 많은 움직임 벡터를 선택한다. 벡터 탐색 기법으로는 VSS 알고리즘이 있다[4].

### 3. WBVC(Weight-Based Vector Composition)

2장에서는 중첩 영역 또는 활동 정보량을 이용하는 대표적인 움직임 벡터 선택 기법인 FDVS와 ADVS 기법에 대해 설명한다. 두 기법은 움직임 벡터의 재구성의 계산량을 크게 줄일 수 있지만, 몇 가지 문제점을 안고 있다. 즉, 이웃한 매크로블록의 중첩 영역의 크기나 활동 정보량의 편차가 작을 경우에는 적절한 움직임 벡터를 선택하기 어렵다는 것이다. 또한 움직임 벡터는 매크로블록 단위로 지정되는데 반해 ADVS 기법에서는 매크로블록이 아닌 블록 단위의 활동량을 계산하므로 잘못된 결과를 초래할 수 있다.

제안하는 WBVC 기법은 위의 두 가지를 함께 고려하여 움직임 벡터를 합성한다. 중첩 영역의 면적을 픽셀 단위로 정확하게 계산하고, 활동 정보량은 중첩된 각각의 매크로블록 단위의 0이 아닌 DCT 계수의 개수를 사용한다. [그림 3]의 예를 바탕으로 한 가중치 계산 및 움직임 벡터의 합성을 위한 단계는 다음과 같다.

**Step 1.** 각 매크로블록의 활동 정보량 계산

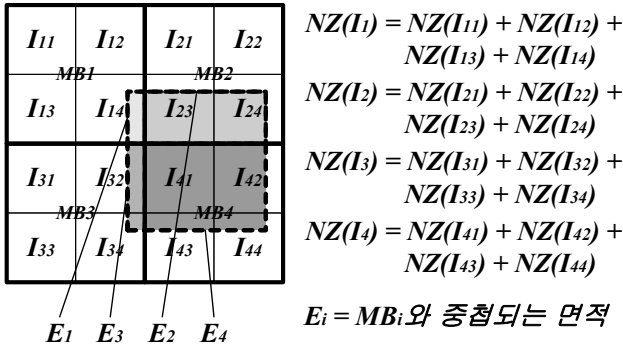
$$E_i * NZ(I_i)$$

Step 2. 가중치 계산

$$W_i = (E_i * NZ(I_i)) / \sum_{i=1}^4 E_i * NZ(I_i)$$

Step 3. 움직임 벡터 합성

$$Composed\ MV = \sum_{i=1}^4 MV_i * W_i$$



[그림 3] WBVC 기법의 가중치

4. 실험 및 성능 평가

본 장에서는 제안한 WBVC 기법의 성능을 평가하기 위해서 아래 표의 인자를 적용하여 실험 결과를 도출했다. 또한, 보다 정확한 성능 평가를 위해서 Full Motion Estimation, FDVS 그리고 ADVS 이외에 제안한 WBVC와 같은 가중치를 사용하여 움직임 벡터를 선택하는 WBVS (Weight-Based Vector Selection) 기법도 비교 대상으로 사용하였다.

[표 1] 입력 비디오 스트림

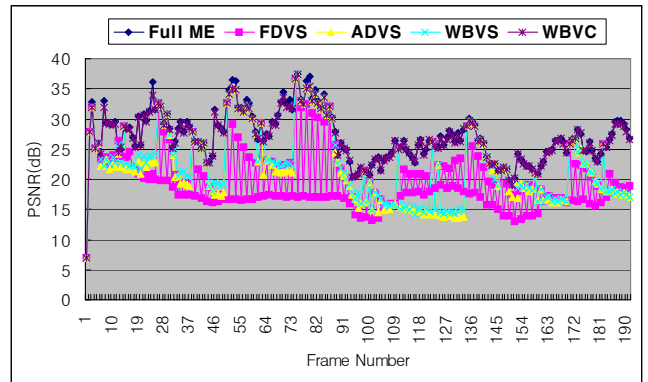
입력 비디오 이름	양자화 파라미터	평균 비트율 (Kbps)	Video Type	프레임률 (fps)	프레임 개수
Carphone	3	310	QCIF	25	382
Foreman		300			400

[표 2] 실험 환경

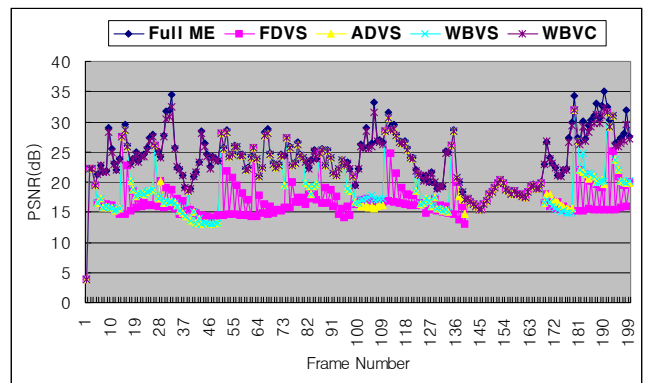
실험 시스템	Intel Pentium IV 800Mhz Dual 512 RAM
비디오 포맷	H.263(IPPP... 로 구성)
비디오 코덱	FFmpeg 0.4.8
출력 비디오 프레임률	12.5 fps
탐색 윈도우 크기	± 2 픽셀 단위

실험에 사용된 비디오 파일은 H.263 QCIF 포맷이고 코덱은 FFmpeg의 H.263 처리부분을 변경하여 각각의 기법들을 적용하였다.

[그림 4], [그림 5]는 "Carphone"과 "Foreman" 비디오 스트림을 입력받아서 각각의 기법을 적용하여 트랜스코딩한 결과를 PSNR 값으로 보여준다. 실험에 사용된 스트림 중 상대적으로 움직임이 더 많은 Foreman 비디오에서도 벡터를 합성하는 것이 좋은 성능을 보인다.

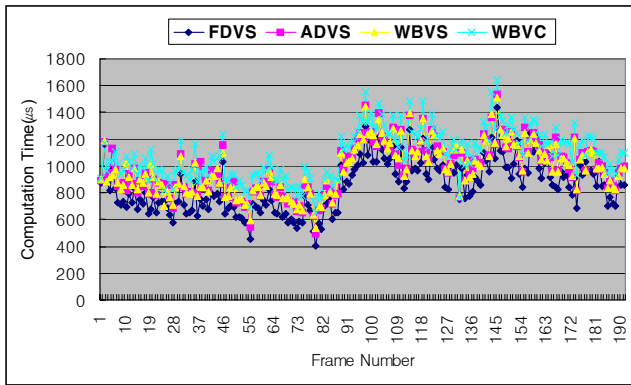


[그림 4] Carphone 비디오의 PSNR 비교

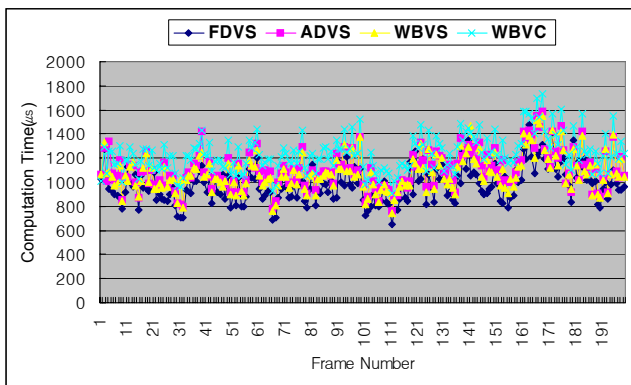


[그림 5] Foreman 비디오의 PSNR 비교

[그림 6], [그림 7]에서는 위와 동일한 비디오로 실험하여 실험한 결과인 기법별 계산 복잡도를 보여준다. Full ME의 경우는 움직임 예측 과정으로 움직임 벡터를 재사용하는 기법들의 계산량보다 18~25배 정도 많기 때문에 그림의 비교 항목에서 제외되었다. 제안한 WBVC 기법의 계산량이 가장 높고, FDVS 기법의 계산량이 가장 적음을 알 수 있다. FDVS와 ADVS의 계산량이 WBVC의 그것과 크게 차이 나지 않고 비슷한 이유는 벡터 탐색 기법인 HAVS와 VSS가 각각 적용되었기 때문이다.



[그림 6] Carphone 비디오의 계산량 비교



[그림 7] Foreman 비디오의 계산량 비교

#### 4. 결론 및 향후 연구 과제

시간당 프레임의 개수를 조절하는 트랜스코딩에서는 새로운 움직임 벡터를 재 추출하지 않고 제거되는 프레임의 움직임 벡터를 이용함으로써 프레임간의 시간적 연관성을 획득할 수 있다. 그러나 움직임 벡터의 정확한 재구성을 위해 본 논문에서는 가중치 기반의 움직임 벡터 합성(WBVC) 기법을 제안하였다. WBVC는 활동 상태 정보량과 중첩 영역의 크기를 동시에 고려한 가중치를 중첩되는 매크로블록 각각에 적용하여 움직임 벡터를 합성하는 기법이다.

실험을 통한 성능 분석 결과에서 제안된 WBVC 기법은 활동량만으로 움직임 벡터를 선택하는 기법인 ADVS 기법과 비슷한 계산량에서 Full Motion Estimation과 비슷한 수준의 PSNR 값을 보였다. 또한, 동일한 가중치를 사용한 WBVS 기법보다 더 좋은 성능을 보였다. 따라서 움직임 벡터 재사용에는 움직임 벡터 선택 기법보다 합성 기법이 더 적합함을 알 수 있었다.

본 논문에서 제안한 움직임 벡터 합성(WBVC) 기법은 단방향 예측 프레임이 존재하는 비디오 스트림

에만 적절할 것이다. WBVC를 양방향 예측 프레임이 존재하는 I, B, P 프레임으로 구성된 MPEG등과 같은 포맷에 적용하기 위해서는 프레임간의 참조 및 예측 방향 결정에 관한 정교한 처리가 필요하다. 사용자의 다양한 요구에 맞추어 서비스하기 위해 필요한 확장된 기법은 향후 과제로 남긴다.

#### 참고문헌

- [1] A. Vetro and C. Christopoulos and H. Sun, "Video Transcoding Architectures and Techniques: An Overview," IEEE signal Processing Magazine, ISSN: 1053-5888, Vol. 20, Issue 2, pp. 18-29, March 2003.
- [2] Kai-Tat Fung and Wan-Chi Siu, "DCT-based Video Frame-skipping Transcoder," Proceedings, IEEE International Symposium on Circuits and Systems (ISCAS'03), Vol. II pp. 656-659, Bangkok Thailand, May 2003.
- [3] J. Youn, M. T. Sun and C. W. Lin, "Motion vector refinement for high performance transcoding," IEEE Trans. Multimedia, vol. 1, pp. 30-40, Mar. 1999.
- [4] K. T. Fung, Y. L. Chan and W. C. Siu, "New architecture for dynamic frame-skipping transcoder," in Proc. IEEE Workshop Multimedia signal Processing, Redondo Beach, CA, pp. 616-621, Dec. 1998.
- [5] "Video coding for low bitrate communication," International Telecommunications Union, Geneva, Switzerland, ITU-T Recommendation H.263, 1998.
- [6] <http://ffmpeg.sourceforge.net>.